README.txt

==========

Project Title: Multisource User Review Analysis for Product Design Insights (60.002 Project 2)

Folder: /DAI\_AID/

Overview:

---------

This project pipeline extracts user sentiment and actionable design insights from public YouTube and Reddit comments about three headphone models:

- Apple AirPods Max

- Sony WH-1000XM5

- Sennheiser Momentum 4 Wireless

It combines data collection, preprocessing, sentiment scoring, and feature-level analysis to identify underperforming product aspects and propose targeted design improvements.

Project Workflow:

-----------------

1. DATA COLLECTION (YouTube + Reddit)

-------------------------------------

- YouTube:

> Used the YouTube Data API to search for product review videos using targeted search queries (e.g., "AirPods Max review").

> Scraped top 5 videos per product.

> Collected up to 500 top-level comments per video.

> Captured fields: comment text, like count, video title, and video ID.

- Reddit:

> Used the PRAW API to search Reddit for each product name.

> Collected comments from the top 10 relevant threads per product (subreddit='all').

> Recursively extracted full comment threads including replies.

> Flattened the structure into a tabular format with comment, score, parent ID, depth.

2. DATA CLEANING

----------------

- Each scraper includes a lightweight cleaning step:

> Removes links and tags like [removed]/[deleted]

> Normalizes whitespace

- Emojis, punctuation, and expressive tokens were retained to preserve sentiment value.

3. WEIGHTED SENTIMENT SCORING

-----------------------------

- Applied to all collected comments (YouTube and Reddit).

- Each comment was scored using a weighted formula:

Weighted Score = (Sentiment Polarity × 0.5) + (Normalized Like Score × 0.3) + (Normalized Comment Length × 0.2)

- Sentiment polarity computed using TextBlob.

4. GAP ANALYSIS TABLE

---------------------

- Extracted noun phrases via POS tagging to identify feature mentions (e.g., "comfort", "latency").

- Filtered for valid product features based on a predefined spec list.

- Calculated sentiment averages per (feature, product) pair.

- Gap Score computed as:

Gap Score = (Average Sentiment of Competitors) – (Sentiment of AirPods Max)

- Any feature with a gap > 0.2 was flagged as "underperforming".

5. SENTIMENT MAPPING ON TOP 5 FEATURES

-------------------------------------

- Selected the 5 features with the highest negative sentiment gap:

→ microphone, fit, build, latency, comfort

- Merged all YouTube + Reddit comments per product.

- Plotted average sentiment per feature across the three products.

6. DESIGN INSIGHT GENERATION (RAG)

----------------------------------

- Used Sentence Transformers to generate embeddings for cleaned AirPods Max comments.

- Indexed with FAISS for efficient semantic search.

- Queried each top feature and retrieved 10 most relevant comments.

- Manually analysed retrieved reviews to extract:

> pain points

> unmet user needs

> improvement suggestions

- Structured outputs stored in RAG design insights folder for reference.

Directory Structure:

--------------------

/DAI\_AID/

│

├── extracted\_data/

│ ├── youtube\_reviews/ # Raw YouTube comment CSVs

│ ├── reddit\_reviews/ # Raw Reddit comment CSVs

│ ├── cleaned\_data/ # Cleaned datasets used for analysis

│ └── RAG design insights/ # Final design opportunity summaries

│

├── keys/ # API credentials

│ ├── google\_key.txt

│ └── reddit\_keys.txt

│

├── working\_code/

│ ├── youtube scrapers/ # Separate scrapers for AirPods, Sony, Sennheiser

│ ├── reddit scrapers/ # Separate Reddit scripts per product

│ ├── gap analysis table FINAL.ipynb

│ └── sentiment analysis FINAL.ipynb

Dependencies:

-------------

- Python 3.7+

- pandas

- nltk

- textblob

- matplotlib

- sentence-transformers

- faiss

- openai

- google-api-python-client

- praw

Note:

-----

All notebooks are modular and reusable across products. Data cleaning is embedded directly in each scraper.